New generation of human machine interfaces for controlling UAV through depth based gesture recognition

Tomás Mantecón Carlos R. del-Blanco Fernando Jaureguizar Narciso García

Grupo de Tratamiento de Imágenes, E.T.S.I. de Telecomunicación, Universidad Politécnica de Madrid.

ABSTRACT

New forms of natural interactions between human operators and UAVs (Unmanned Aerial Vehicle) are demanded by the military industry to achieve a better balance of the UAV control and the burden of the human operator. In this work, a human machine interface (HMI) based on a novel gesture recognition system using depth imagery is proposed for the control of UAVs. Hand gesture recognition based on depth imagery is a promising approach for HMIs because it is more intuitive, natural, and non-intrusive than other alternatives using complex controllers. The proposed system is based on a Support Vector Machine (SVM) classifier that uses spatio-temporal depth descriptors as input features. The designed descriptor is based on a variation of the Local Binary Pattern (LBP) technique to efficiently work with depth video sequences. Other major consideration is the especial hand sign language used for the UAV control. A tradeoff between the use of natural hand signs and the minimization of the inter-sign interference has been established. Promising results have been achieved in a depth based database of hand gestures especially developed for the validation of the proposed system.

Keywords: UAV control, gesture recognition, LBP, depth imagery, SVM.

1. INTRODUCTION

Nowadays, Unmanned Aerial Vehicles (UAVs) are part of both military and civil aviation environments. As a consequence, an increasing investment in resources is being made to ease the control and management of this kind of aerial vehicles. In this line, the work presented in[1] has focused on the control a flying robot that is in the same filed of view as a human operator by means of a gesture-based interface, which tries to simulate the type of interaction that humans have with birds, specifically falconeering. Other works are more concern in commanding teams of UAVs using hand gestures,[2] multimodal interfaces,[3] or artificial cognition.[4]

One of the most promising Human Machine Interaction (HMI) paradigms for the UAV commanding is the one offered by the field of computer vision. There is a wide range of works that have proposed different approaches for HMI: based on human-pose recognition, eye-tracking, or hand-gesture recognition. Especially, hand-gesture recognition has generated large expectations for HMI to alleviate the restrictive level of interaction of typical devices such as the keyboard or the mouse. The kind of input information is also another key part in HMI applications. Most of the works have used color information for this task,[5–7] although recently depth data is becoming more and more popular for the recognition of hand gestures.[8–10] This has been possible thanks to the appearance of the Microsoft Kinect 1,[11] a low-cost depth camera that has spread the use of the depth information in computer vision. However, this device poses some problems for an accurate depth-based hand gesture recognition system: a reduced resolution in depth, high level of noise, and missing data.[12] These problems makes almost impossible to extract reliable features from the depth relief of different hand/finger poses. Currently, a second generation of low-cost depth sensors have arrived with improved depth/range resolution, allowing to design new algorithms that take advantage of improved quality in depth resolution. Kinect 2[13] from Microsoft (the second version of Kinect 1) forms part of this new generation of depth cameras.

In this paper, a new hand gesture recognition algorithm based on depth imagery is proposed for the commanding of UAVs. The recognition algorithm is based on a machine learning framework that uses a Support Vector Machine (SVM) classifier to recognize a predefined set of hand gesture actions. The key contribution is a novel descriptor, called Array of Spatio-Temporal Histograms of Local Quantized Depth Patterns (ASTH-LQDP),

Emails: {tmv,cda,fjn,narciso}@gti.ssr.upm.es

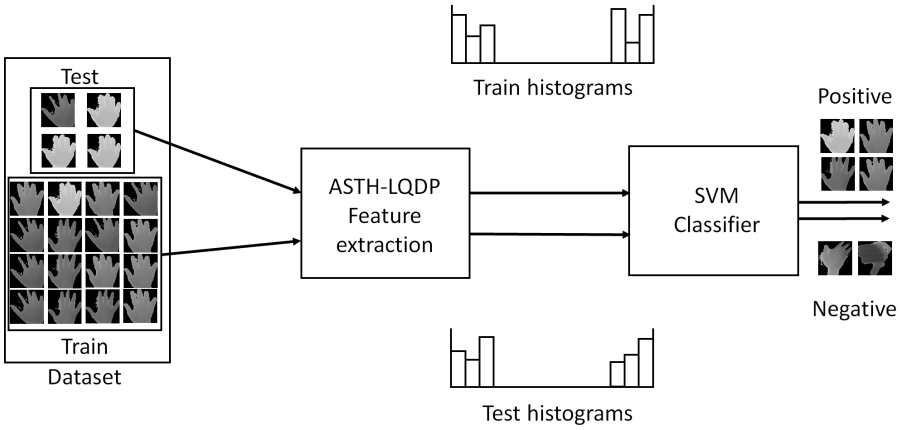


Figure 1. Machine learning framework for the recognition of hand gestures in depth imagery.

which efficiently encodes the spatio-temporal depth patterns for a robust recognition. The core of this descriptor is based on the popular Local Binary Pattern (LBP) operator, but largely extended to be more discriminant and robust for video depth sequences. One of the most important design steps is a quantification stage that exploits the extended resolution in depth that the new generation of low-cost depth cameras can offer, such as Kinect 2. This fact strongly contrasts with the standard version of LBP that discards such information. Fig.1 illustrate the proposed hand gesture recognition algorithm. In addition, a depth-based hand gesture database acquired by the new Kinect 2 sensor has been created (which is freely and publicly available in[14]) to properly validate the proposed solution.

The organization of the paper is as follows. Section 2 describes different variants of the popular LBP descriptor for its use in color and depth information. A description of the proposed feature descriptor is presented in Section 3. The application of a set of SVM classifiers for the hand gesture recognition is described in Section 4. The obtained results with the proposed algorithm are presented in Section 5. Finally, conclusions are drawn in Section 6.

2. LBP FEATURES IN COLOR AND DEPTH IMAGERY

2.1 COLOR VERSUS DEPTH IMAGERY

The problems that a hand gesture recognition application has to face when uses color or depth imagery are partially different. One of the most challenging problems that only appears in color imagery is related with the illumination: shadows, reflections and illumination changes. Although a lot of different techniques and approaches exist to deal with these problems, it is difcult to successfully solve all of them, especially when all those phenomena can simultaneously occur. Consequently, the recognition rate can decrease considerably.

Unlike color imagery, depth data acquired by active depth sensors are immune to the previous problems, at least in indoor scenarios. Even more, they can perfectly operate in the absence of illumination.[15] In the case of outdoor scenarios, active depth sensors are sensitive to direct effect of sunlight, which could blind them.

Other key difference between color and depth imagery is the spatial resolution. Color cameras can have high resolutions, ranging from 640×480 pixels up to more than 1920×1080. However, the resolution of active depth sensors is significantly less, and only a few models reach a resolution of 640×480 pixels. Some of them some, based on the recognition of a structure light pattern, have become very popular because their relative low price. An example is the Microsoft Kinect 1. However, the effective resolution in depth is less than other much more expensive depth cameras. Recently, new low-cost depth sensors based on the time-of-flight technology have appeared with an increased depth resolution. Two examples are: the new Microsoft Kinect 2 and the Creative Senz3D, which has a resolution of 640×480 and 320×240 pixels, respectively.

On the other hand, color and depth imagery share common problems for the recognition task such as variations in scale, location, orientation (inplane rotation), and pose (out-of-plane rotation), occlusions, and facial expressions.

2.2 LBP BASED FEATURES IN COLOR IMAGERY

Local Binary Pattern (LBP) has become a popular descriptor due to its robustness to dramatic illumination changes, its computational efficiency, and the good results achieved in several tasks, such as texture and face classification. This operator calculates a binary pattern by thresholding the neighborhood of each pixel, and converting the resulting binary number into a decimal one. Then, a histogram is generated from all the computed decimal numbers, which represents the feature vector.

The original LBP operator was designed for texture description.[16] It thresholds a 3x3 neighborhood by the intensity value of the center pixel. I.e., only the sign difference between the values of the neighbors and the value of the center pixel are taken into account. As a result, an 8 bit binary number is generated from the concatenation of all the sign differences, which in turn is converted to a decimal-number (a label) that represents the neighborhood pattern. The last step is to compute a 2^8-bin histogram with all the pattern labels obtained from an image region.

Several variations of the LBP operator have appeared since then. A multi-resolution extension of the standard LBP operator was proposed in,[17] which was able to use neighborhoods of different sizes. The neighborhood was defined as a circle of radius R, where a set of P equidistant sampling points were taken. Other popular variations of the LBP descriptor were focused on reducing the dimensionality of the resulting feature vector, filtering out those pattern labels that were infrequent.[18] With the purpose of collecting more information from the pixel neighborhood, the Complete Local Binary Pattern (CLBP) was proposed, which uses a Local Difference Sign-Magnitude Transformation (LDSMT).[19] This transformation decomposes the differences into two components: the sign and the magnitude. While the sign component is equivalent to the conventional LBP, the magnitude component may provide more discriminative information. In addition, the intensity value of the center pixel is also proposed to be used. The combination of these three components can achieve an important improvement in the subsequent classification task.

The LBP feature descriptor and its variations share a common limitation, they do not provide spatial information, since the generated histogram takes only into account the pattern occurrences, discarding the spatial location from which they were computed.

2.3 LBP BASED FEATURES IN DEPTH IMAGERY.

The LBP descriptor have been recently applied for depth imagery in face recognition applications. In,[20] a multi-scale LBP have been used to describe the 3D facial features. In,[21] a 3D model is used to extract LBP features, instead of real depth imagery. In,[22] a variation of the LBP is proposed, called Local Normal Binary Pattern (LNBP), consisting in using normal vectors instead of intensity values. I.e., this descriptor computes the neighborhood differences among the values of normal vectors, instead of depth values.

The LBP descriptor has been also used in depth imagery for the task of hand gesture recognition. In,[23]the LBP features extracted from depth and color images are combined to improve the recognition rate. And in[24] the SIFT descriptor is combined with the LBP one to augment the feature vector and increase the recognition capability.

However, the previous methods simply apply the LBP descriptor or one of its variation to depth imagery, without considering the existing differences between depth and color imagery. A kind of exception is the work presented in,[22] which uses normal vectors to locally represent depth patches.

3. ASTH-LQDP DESCRIPTOR

The proposed Array of Spatio-Temporal Histograms of Local Quantized Depth Patterns (ASTH-LQDP) is a high discriminative descriptor for depth video sequences that is able to efficiently encode hand gestures for recognition purposes. It is radical evolution from the popular Multi-Scale LBP justified by two facts: the use of depth imagery, instead of color, and the introduction of the temporal dimension, needed to encode hand

gestures. Regarding the first issue, the ASTH-LQDP descriptor is adapted to the special characteristics of the depth imagery, such as the spatial and depth resolution, and the absence of strong variations in pixel values (unlike the color imagery, where there can be strong variations due to changes in illumination). Related the second issue, the ASTH-LQDP descriptor uses the temporal dimension to increase the recognition capability and to be able to detect dynamic hand gestures that can be executed at different speeds.

The computation of the ASTH-LQDP descriptor can be divided into four stages or levels: neighborhood level, block level, image level, and video sequence level. The first one, the neighborhood stage or level, computes the depth value differences between a reference pixel and its neighborhood. These depth differences are adaptively quantized with N_q bits, which are finally used to generate a decimal code, called Local Quantized Depth Patterns (LQDP), which describes the previous neighborhood. The block level densely computes LQDP codes in a depth image region and generates a histogram with the resulting LQDP codes. The image level spatially divides the image into $N_b \times N_b$ blocks, and stacks up in a row vector the computed LQDP histograms from the blocks. Finally, the video sequence level temporally divides the sequence into N_i consecutive images, and stacks up in a row vector the spatio-temporal LQDP histograms from a subset of the images.

Fig. 2 illustrates the four stages/levels for the computation of the ASTH-LQDP descriptor along with the main tasks performed in everyone.

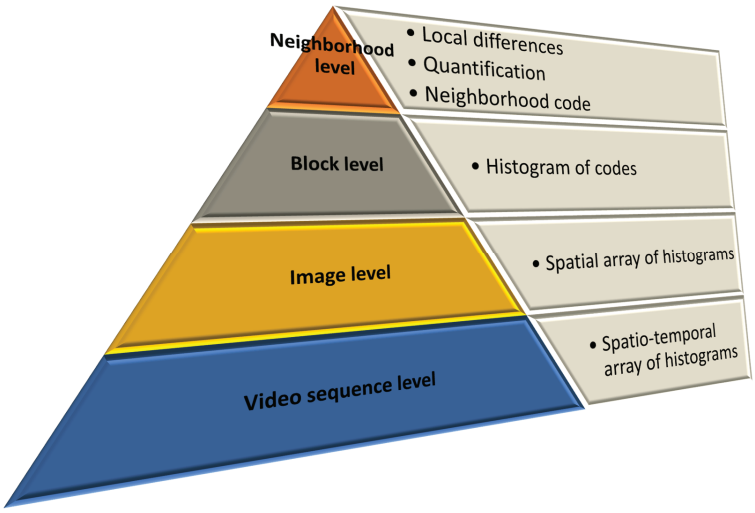


Figure 2. Illustration of the four stages/levels for the computation of the ASTH-LQDP descriptor along with the main tasks performed in everyone.

3.1 Neighborhood level

The neighborhood level can be divided into three stages: the computation of local depth differences between the central pixel and its neighborhood, the quantification of those differences, and the generation of a decimal number code for each pixel of the depth-map.

The first stage, illustrated in Fig. 3, is the computation of differences between the depth values of a central pixel and its neighborhood. This is mathematically represented by the following vector

$$D = [d_0 - d_c, d_1 - d_c, ..., d_{P-1} - d_c], \tag{1}$$

where P is the number of neighborhood samples, d_c is the depth value of the central pixel, and d_i with $i = 0, ..., P-1$ are the depth values of the neighborhood samples.

The second stage uniformly quantizes every value of the vector of depth differences D. For this purpose, N_q bits are used to encode the sign and the magnitude of the difference, where the sign is encoded as the most

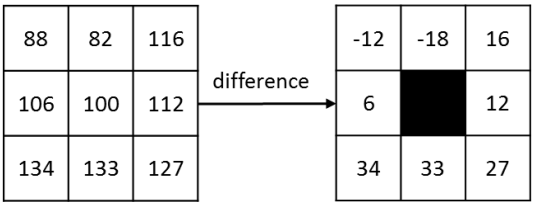


Figure 3. Example of the computation of local depth differences in a neighborhood of $P = 8$ pixels.

significant bit. Since the range of depth values is relatively large (typically 2^{12} values), only a subset of the range of difference values will be quantized. This subset is chosen to cover the most significant differences in the recognition application. In the case of hand gesture recognition, the subset will contain depth information of the hand relief. This is achieved by imposing a fix width for the quantification intervals, determined by the parameter Δ. Fig. 4 illustrates the process. Then, a binary number is generated by concatenating the binary number resulting from the quantification of every component in D.

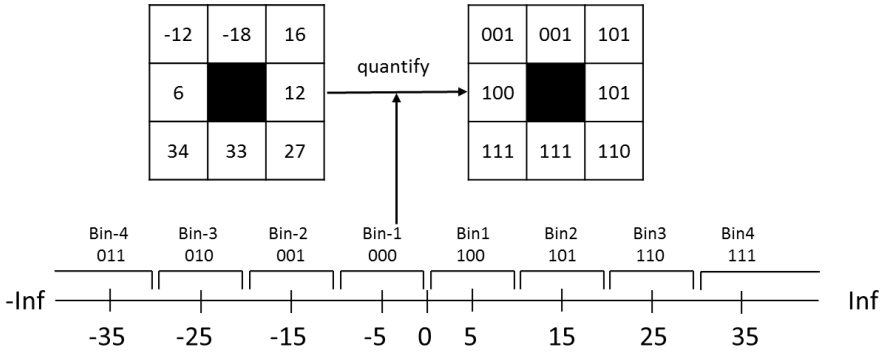


Figure 4. Example of the quantification process of the vector of local differences D. A number of $N_q = 3$ bits and a value of $\Delta = 10$ have been chosen for the quantification of every component in D.

Finally, the obtained binary number resulting from the quantification process is converted into a decimal number, called Local Quantized Depth Pattern (LQDP), which is used as a label to encode the depth pattern of the neighborhood of a pixel. Fig. 5 illustrates the process.

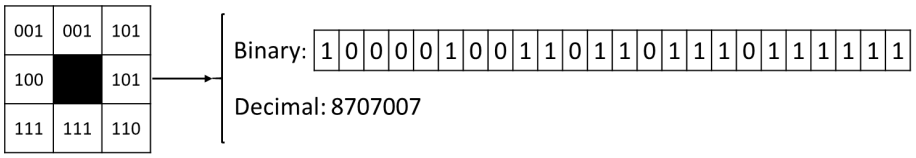


Figure 5. Computation of the neighborhood label, which is a decimal number obtained from the binary code obtained in the quantification stage.

3.2 Block, image, and video sequence levels

The block level computes the LQDP codes in an image region, and then calculates a histogram of LQDP codes, called H-LQDP, which is used as feature vector to characterize the corresponding image region. The H-LQDP histogram is composed by $2^{N_q \times P}$ bins. The selection of the parameters N_q and P is critical to restrict the length of the resulting histogram.

The image level divides the image into $N_b \times N_b$ different block regions, and applies the block level stage to each block region to obtain a H-LQDP. Then, a new feature vector is obtained by concatenating all the H-LQDP

histograms, which is called Array of Spatial Histograms of Local Quantized Depth Patterns (ASH-LQDP). This division of the depth image into blocks allows to incorporate some spatial information to the H-LQDP feature descriptor, which lacked of any spatial reference due to the histogram computation of LQDP codes. Fig. 6 illustrates the block and image levels.

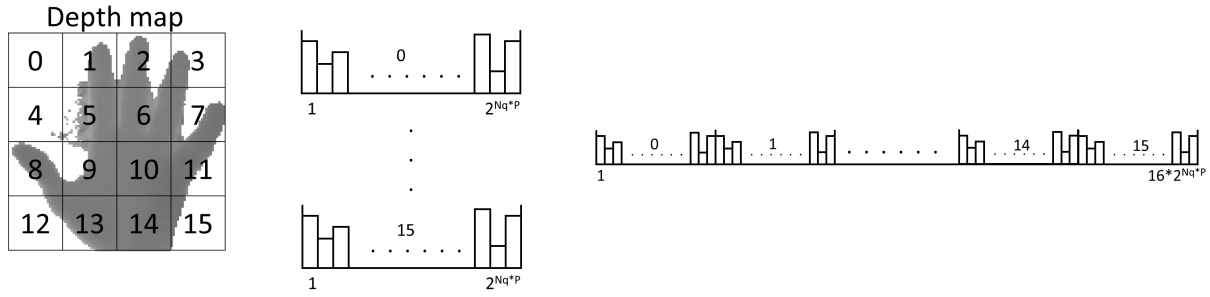


Figure 6. Illustration of the block and image levels. A histogram of LQDP codes is computed for each block of the image, and then a new feature vector is computed as the concatenation of all the histograms resulting from each block.

The video sequence level extends the ASH-LQDP descriptor to be able to recognize dynamic hand gestures that spans temporally. In this level, a temporal subsampling of N_i frames in the video sequence is performed, resulting in a selection of depth images. Then, an ASH-LQDP descriptor is computed for every depth frame. Finally, a new feature vector is obtained as the temporal concatenation of the N_i ASH-LQDP descriptors of the subset of sampled images. This vector is called Array of Spatio-Temporal Histograms of Local Quantized Depth Patterns (ASTH-LQDP). The temporal subsampling has a dual purpose. On the one hand, it allows to recognize hand gestures executed at different speeds. And on the other hand, the subsampling shortens the resulting feature vector in comparison with the exhaustive version that uses of all the frames in the sequence. Since the length of the ASTH-LQDP descriptor is equal to $N_i \times N_b \times N_b \times 2^{N_q \times P}$, which can give rise to a prohibitive feature sizes for its used in practical machine learning systems, the subsampling procedure is very important. Fig. 7 illustrates the video sequence level.

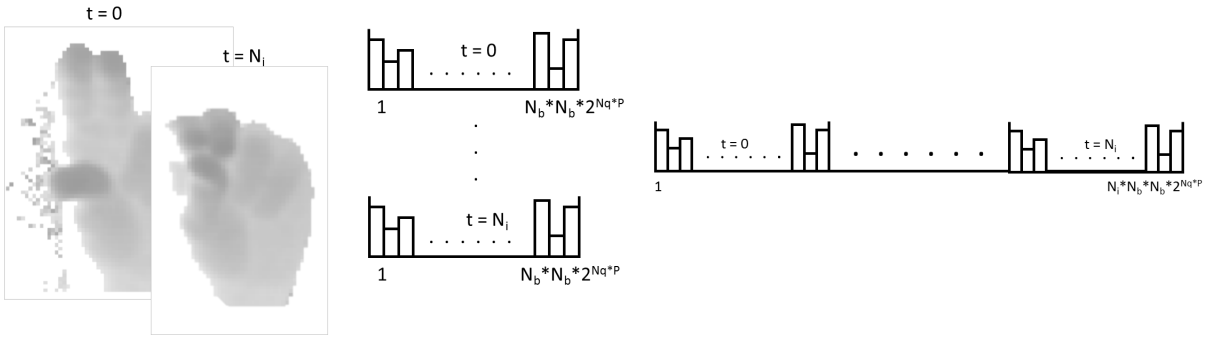


Figure 7. Computation of the ASTH-LQDP feature vector at the video sequence level.

4. SVM BASED RECOGNITION

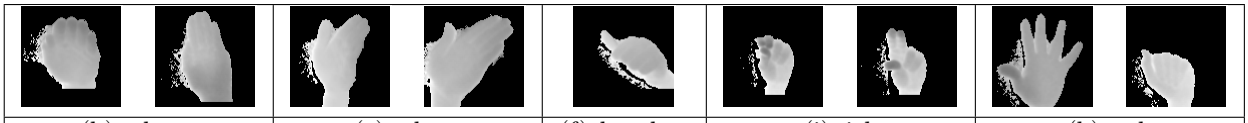
A binary Support Vector Machine (SVM) classifier is used as basis for the recognition task. As the purpose is to recognize a set of G different hand gestures, a set of G SVM classifiers are trained to recognize every gesture following a one-versus-all strategy. A Hellinger kernel, more commonly known as Bhattacharyya coefficient,[25] has been used for each SVM, which is mathematically represented as

$$k(f, f') = \sum_i \sqrt{f(i)f'(i)}, \tag{2}$$

where f and f' are the ASTH-LQDP based feature vectors of the test and training databases, respectively. The positive training samples for each hand gesture is composed by the ASTH-LQDP feature vectors extracted from the video sequences related to that gesture, and the negative ones are the ASTH-LQDP feature vectors extracted from those video sequences related to the other hand gestures. The same number of positive and negative training samples are used to train each SVM classifier.

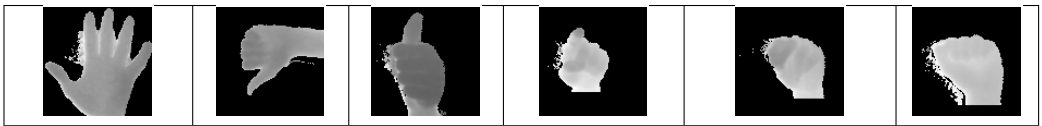
5. EXPERIMENTAL RESULTS

The proposed hand-gesture recognition algorithm, from now on called ASTH-LQDP-SVM, has been tested and compared with other state-of-the-art algorithms. For that purpose, a depth-based hand-gesture database has been generated (Figs. 8 and 9) using the Microsoft Kinect 2, which can be downloaded from.[14] The database is composed by 11 different hand gestures, 5 of them are dynamic (Fig. 8) and other 6 are static (Fig. 9). Each gesture was performed by 6 different subjects (3 men and 3 women), whose hands are of different sizes. A total number of 100 frames were recorded per gesture and per person, resulting in 600 depth frames available for each gesture. The regions containing the hand-gestures were manually selected using a bounding box of 100×100 pixels. The database was divided into two groups: the training group formed by the 80% of the database, and the testing group formed by the remaining 20%.



(b)palm$_{ud}$	(c)palm$_{lr}$	(f)thumb$_{ud}$	(i)pick	(k)grab

Figure 8. Dynamic hand gestures of the database: (b) open palm moving up and down, (c) open palm moving from left to right, (f) close hand with the thumb extended and moving up and down, (i) moving three fingers like piking something, and (k) open and close the hand like grabbing something. The gray level of those depth-maps represents distances between 600 mm (darker) and 700 mm (lighter).



(a)open palm	(d)thumb$_d$	(e)thumb$_u$	(g)one finger	(h)three fingers	(j)fist

Figure 9. Static hand gestures of the database: (a) static palm open, (d) close hand with the thumb pointing down, (e) close hand with the thumb pointing up, (g) one finger pointing to the camera, (h) three joint fingers pointing to the camera and (j) close hand (fist). The gray level of those depth-maps represents distances between 600 mm (darker) and 700 mm (lighter).

The ASTH-LQDP-SVM algorithm has been compared with the LBP algorithm using the the confusion matrix (CM) as metrics. The confusion matrix is widely used in object recognition to measure the recognition performance. Each column of the matrix represents the number of predicted hand-gestures belonging to each class, and each row represents the total number of outcomes (positives and negatives) that belongs to each class. Each element of the main diagonal of that matrix represents the accuracy of each class, which is expressed as follows

$$\text{Accuracy} = 100 \times \frac{\text{Total number of correct hand gestures of a class}}{\text{Total number of hand gestures of a class}} \tag{3}$$

The used configuration to compute the proposed descriptor ASTH-LQDP is: each depth image was divided into 4×4 blocks with 25×25 pixels each one; the number of neighbors is $P = 4$ in a west, north, east and south spatial arrangement; the number of samples for the temporal subsampling is $N_i = 3$; the number of bits for the quantization is $N_q = 3$; and the length of each interval is $\Delta = 10$mm. The parameters have been

chosen as a tradeoff between the length of the resulting feature vector (related to the computational cost) and its discriminative capacity. With the proposed configuration, it is possible to distinguish differences of 10mm in a range between −40mm and 40mm, which is well adapted to the hand gesture recognition task, as the Fig. 10 shown, where the maximum relief is less than 50mm. Regarding the standard LBP algorithm, two different configurations have been used: one uses $P = 8$ neighbors (LBP_8), and the other uses $P = 4$ neighbors (west, north, east and south) (LBP_4). The rest of common parameters, such as the number of blocks per image, is the same.

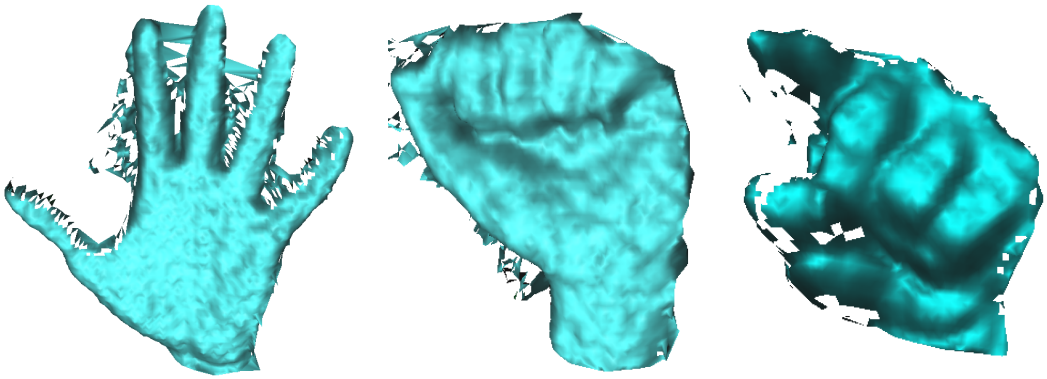


Figure 10. Meshed hands resulting from the point clouds acquired by the Kinect 2. Three different gestures are shown, where some noise artifacts can be observed.

The results of the proposed ASTH-LQDP algorithm are shown in Table 1. Each row and each column represents each class (hand gesture) represented in Figs. 8 and 9. As it can be observed, all of them have an accuracy value greater than 90%, except for the classes $thumb_d$ and $thumb_u$ that are confused with class $thumb_{ud}$. To increase the recognition accuracy of the system, the symbol (f)$thumb_{ud}$ of higher inter-sign interference is removed from the final hand gesture vocabulary. The new results of that new filtered hand gesture vocabulary are presented in Table 2. Under this situation, the accuracy is greater than 90%, specifically the classes $thumb_d$ and $thumb_u$ have increased their accuracy from 85.83% to 92.50%, and from 86.67% to 100%, respectively.

Table 1. Results of the ASTH-LQDP algorithm using 11 classes.

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
(a)	97,50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	2,50
(b)	0,00	94,17	5,00	0,00	0,00	0,00	0,00	0,83	0,00	0,00	0,00
(c)	0,00	0,00	92,50	0,00	0,00	7,50	0,00	0,00	0,00	0,00	0,00
(d)	0,00	0,00	0,00	85,83	0,00	14,17	0,00	0,00	0,00	0,00	0,00
(e)	0,00	0,00	0,00	0,00	86,67	13,33	0,00	0,00	0,00	0,00	0,00
(f)	0,00	0,00	0,83	1,67	0,00	97,50	0,00	0,00	0,00	0,00	0,00
(g)	0,00	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00
(h)	0,00	0,00	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00
(i)	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00
(j)	0,00	0,00	3,33	0,00	0,00	0,00	0,00	0,00	0,00	96,67	0,00
(k)	0,00	8,33	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1,67	90,00

Tables 3, 4, and 5 show the results using 10 classes for different algorithms and configurations to be compared with the results already shown in Table 2. Table 3 shows the results of the ASTH-LQDP algorithm but without temporal information (called ASH-LQDP), i.e. only one image is used for the hand gesture recognition, which represents an extreme case of the temporal subsampling process. Tables 4 and 5 show the results of the LBP algorithm using 8 and 4 neighbors, respectively. The best results in terms of recognition accuracy are obtained

Table 2. Results of the ASTH-LQDP algorithm using 10 classes after the inter-sign interference criterion is applied.

	(a)	(b)	(c)	(d)	(e)	(g)	(h)	(i)	(j)	(k)
(a)	97,50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	2,50
(b)	0,00	93,33	5,83	0,00	0,00	0,00	0,83	0,00	0,00	0,00
(c)	0,00	0,00	92,50	0,00	7,50	0,00	0,00	0,00	0,00	0,00
(d)	0,00	0,00	0,00	92,50	7,50	0,00	0,00	0,00	0,00	0,00
(e)	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00	0,00
(g)	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00
(h)	0,00	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00
(i)	0,00	0,00	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00
(j)	0,00	0,00	3,33	0,00	0,00	0,00	0,00	0,00	96,67	0,00
(k)	0,00	9,17	0,00	0,00	0,00	0,00	0,00	0,00	1,67	89,17

with the proposed solution with temporal information. Making the comparison with the results in Table 3, it can be claimed that adding temporal information achieves better accuracy results, specially in the case of the grab gesture that is confused with palm or fist gestures. The results obtained by the LBP algorithm in both configuration are also less accurate than the ones obtained by the presented ASTH-LQDP algorithm.

Table 3. Results of the ASH-LQDP algorithm using 10 classes (similar to the ASTH-LQDP algorithm but without temporal information, i.e. only one image is used for the hand gesture recognition).

	(a)	(b)	(c)	(d)	(e)	(g)	(h)	(i)	(j)	(k)
(a)	91,67	0,00	0,00	0,00	0,00	2,50	0,00	0,00	5,00	0,83
(b)	5,00	93,33	1,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
(c)	0,00	0,00	84,17	1,67	11,67	1,67	0,00	0,00	0,00	0,83
(d)	0,00	0,00	0,00	92,50	7,50	0,00	0,00	0,00	0,00	0,00
(e)	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00	0,00
(g)	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00
(h)	0,00	0,00	0,00	0,00	0,00	0,00	97,50	0,00	2,50	0,00
(i)	0,00	0,00	0,00	0,00	3,33	0,00	1,67	95,00	0,00	0,00
(j)	0,00	0,00	3,33	1,67	0,00	0,00	0,00	0,00	95,00	0,00
(k)	2,50	16,67	0,00	0,00	0,00	2,50	0,00	0,00	8,33	70,00

Table 4. Results of the LBP algorithm with 8 neighbors algorithm using 10 classes.

	(a)	(b)	(c)	(d)	(e)	(g)	(h)	(i)	(j)	(k)
(a)	91,67	0,00	0,00	0,00	0,00	2,50	0,00	0,00	5,00	0,83
(b)	6,67	85,83	0,83	0,00	0,00	0,00	0,00	0,00	0,00	6,67
(c)	0,00	0,00	88,33	0,83	7,50	1,67	0,00	0,00	0,83	0,83
(d)	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00	0,00	0,00
(e)	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00	0,00
(g)	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00
(h)	0,00	0,00	0,00	0,00	0,00	0,00	97,50	0,00	2,50	0,00
(i)	0,00	0,00	0,00	0,00	3,33	0,00	5,83	90,83	0,00	0,00
(j)	0,00	0,00	5,00	0,00	0,00	0,00	0,00	0,00	95,00	0,00
(k)	2,50	16,67	0,00	0,00	0,00	2,50	0,00	0,00	12,50	65,83

Finally, Table 6 shows a summary of global measures. As it can be seen, removing the gesture $thumb_{ud}$,

Table 5. Results of the LBP algorithm with 4 neighbors algorithm using 10 classes.

	(a)	(b)	(c)	(d)	(e)	(g)	(h)	(i)	(j)	(k)
(a)	92,50	0,00	0,00	0,00	0,00	2,50	0,00	0,00	5,00	0,00
(b)	13,33	72,50	2,50	0,00	1,67	0,00	0,00	0,00	5,00	5,00
(c)	1,67	0,83	74,17	2,50	17,50	1,67	0,00	0,00	0,00	1,67
(d)	0,00	0,00	0,00	92,50	7,50	0,00	0,00	0,00	0,00	0,00
(e)	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00	0,00
(g)	0,00	0,00	0,00	0,00	0,00	100,00	0,00	0,00	0,00	0,00
(h)	0,00	0,00	0,00	0,00	0,00	1,67	81,67	12,50	4,17	0,00
(i)	0,00	0,00	1,67	0,00	3,33	0,00	2,50	92,50	0,00	0,00
(j)	0,00	2,50	5,00	0,00	0,00	0,00	0,00	0,00	92,50	0,00
(k)	10,83	27,50	0,00	0,00	0,00	2,50	0,00	3,33	10,00	45,83

increase the mean accuracy of all hand gestures from 94.62% to 96.17%. It can be noticed that the presented solution is the one that achieves better results in mean accuracy, and also it is the one with the minimum standard deviation of the accuracy.

Table 6. Mean accuracy and standard deviation foo the compared algorithms.

	ASTH-LQDP	ASTH-LQDP	ASH-LQDP	LBP_8	LBP_4
	11 gestures	10 gestures	10 gestures	10 gestures	10 gestures
Mean accuracy (%)	94.62	96.17	91.92	91.50	84.42
Standard deviation	5.24	4.01	8.97	10.36	16.63

6. CONCLUSIONS

An accurate hand-gesture recognition system for human machine interface (HMI) applications has been presented with the aim of controlling UAVs. Using depth imagery allows a more intuitive, natural and non-intrusive interaction than the other more complex systems. The key of the systems is a new hand gesture descriptor adapted to the depth data provided by the new Kinect 2 sensor, which has been used in a recognition system based on a set of SVMs. Excellent classification results have been obtained, outperforming other approaches of the literature.

ACKNOWLEDGMENTS

This work has been partially supported by the Ministerio de Economía y Competitividad of the Spanish Government under project TEC2010-20412 (Enhanced 3DTV) and by Airbus Defence and Space under project SAVIER, Open Innovation Program.

REFERENCES

[1] W. S. Ng and E. Sharlin, "Collocated interaction with flying robots," in *20th IEEE International symposium on robot and human interactive communication*, pp. 143–149, 2011.

[2] V. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori, "Hri in the sky: Creating and commanding teams of uavs with a vision-mediated gestural interface," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 617–623, 2013.

[3] G. Jones, N. Berthouze, R. Bielski, and S. Julier, "Towards a situated, multimodal interface for multiple uav control," in *IEEE International Conference on Robotics and Automation*, pp. 1739–1744, 2010.

[4] C. Meitinger and A. Schulte, "Human-uav co-operation based on artificial cognition," in *Engineering Psychology and Cognitive Ergonomics*, D. Harris, ed., **5639**, pp. 91–100, Springer Berlin Heidelberg, 2009.

[5] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* **37**(3), pp. 311–324, 2007.

[6] Y. Ding, H. Pang, X. Wu, and J. Lan, "Recognition of hand-gestures using improved local binary pattern," in *International Conference on Multimedia Technology*, pp. 3171–3174, 2011.

[7] D. Wickeroth, P. Benolken, and U. Lang, "Markerless gesture based interaction for design review scenarios," in *Second International Conference on the Applications of Digital Information and Web Technologies*, pp. 682–687, 2009.

[8] L. Lin, Y. Cong, and Y. Tang, "Hand gesture recognition using rgb-d cues," in *International Conference on Information and Automation*, pp. 311–316, 2012.

[9] M. Van den Bergh and L. Van Gool, "Combining rgb and tof cameras for real-time 3d hand gesture interaction," in *IEEE Workshop on Applications of Computer Vision*, pp. 66–72, 2011.

[10] D. Minnen and Z. Zafrulla, "Towards robust cross-user hand tracking and shape recognition," in *IEEE International Conference on Computer Vision Workshops*, pp. 1235–1241, 2011.

[11] "Microsoft Corporation. Kinect for Xbox 360 http://dx.doi.org/10.1007/s10107-010-0420-4."

[12] M. Camplani, T. Mantecon, and L. Salgado, "Depth-color fusion strategy for 3-d scene modeling with kinect," *IEEE Transactions on Cybernetics* **43**(6), pp. 1560–1571, 2013.

[13] "Microsoft Corporation. Kinect for Xbox ONE http://www.microsoft.com/en-us/kinectforwindows/."

[14] "Depth based gesture recognition database https://sites.google.com/site/depthgestrecog/."

[15] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review," *IEEE Transactions on Cybernetics* **43**(5), pp. 1318–1334, 2013.

[16] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition* **29**(l), pp. 51–59, 1996.

[17] T. Ojala, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(7), pp. 971–987, 2002.

[18] S. Liao, M. Law, and A. Chung, "Dominant local binary patterns for texture classification," *IEEE Transactions on Image Processing* **18**(5), pp. 1107–1118, 2009.

[19] Z. Guo and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions on Image Processing* **19**(6), pp. 1657–1663, 2010.

[20] D. Huang, M. Ardabilian, Y. Wang, and L. Chen, "3-d face recognition using elbp-based facial description and local feature hybrid matching," *IEEE Transactions on Information Forensics and Security* **7**(5), pp. 1551–1565, 2012.

[21] H. Tang, B. Yin, Y. Sun, and Y. Hu, "3d face recognition using local binary patterns," *Signal Processing* **93**(8), pp. 2190 – 2198, 2013.

[22] G. Sandbach, S. Zafeiriou, and M. Pantic, "Local normal binary patterns for 3d facial action unit detection," in *19th IEEE International Conference on Image Processing*, pp. 1813–1816, 2012.

[23] C. Weerasekera, M. Jaward, and N. Kamrani, "Robust asl fingerspelling recognition using local binary patterns and geometric features," in *International Conference on Digital Image Computing: Techniques and Applications*, pp. 1–8, 2013.

[24] X. Zhu and K. Wong, "Single-frame hand gesture recognition using color and depth kernel descriptors," in *21st International Conference on Pattern Recognition*, pp. 2989–2992, 2012.

[25] E. Choi and C. Lee, "Feature extraction based on the bhattacharyya distance," *Pattern Recognition* **36**(8), pp. 1703 – 1709, 2003.